EXHIBIT 3
Part 4 of 4

as, for example SHOW, ENABLE or LOGOUT." *Id.* at 8:53-65. "The text is supplied by the single attribute 'text'." *Id.* at 8:65-67. Through this example, Froyd provides an illustration of an XML tag with an XML parameter ("text") indicating that the XML tag includes one or more CLI keywords. Clark Decl. ¶ 88. A POSA would understand that, as part of converting from XML to CLI, the command would be parsed to identify the XML command attribute. *Id.*

3. [3C] "traversing the input after the XML command attribute to identify any keywords and any parameters associated with the XML command attribute;"

Gorthy in combination with Froyd discloses this limitation. Clark Decl. ¶ 89. As discussed above with respect to claim 1, for instance, a POSA would understand that the process disclosed in Gorthy of translating the input XML command into a CLI command involves identifying keywords and parameters. *See id.*; *see also* discussion of claim element 1C. In Appendix B, for instance, Gorthy provides an example of an XML configuration schema that includes XML tags that contain XML parameters indicating that the tag contains one or more CLI keywords. *Id.* at ¶ 89. The XML schema disclosed in Gorthy thus shows that the system is able to identify these things. *Id.* A POSA would understand that the translation process involves traversing the input to identify keywords and parameters associated with the XML tag. *Id.*; *see also* discussion of claim element 1C; Ex. 1004 at 8:51-67.

4. [3D] "translating the XML command attribute into the CLI command;"

Gorthy in combination with Froyd discloses this limitation. Clark Decl. ¶ 90; *see also* discussion of claim element 1C. As discussed above, Gorthy discloses a system that translates XML-based configuration commands having CLI syntax to CLI-based commands using an XML schema. *See* discussion of claim element 1C; Ex. 1003 at 3:9-16. The XML configuration schema is used to "reformat the XML-based command into a proper CLI format." Ex. 1003 at 3:15-16. A POSA would understand that this conversion process involves translating XML tags containing XML parameters and CLI keywords into CLI commands. Clark Decl. ¶ 90; *see also* discussion of claim element 1C; Ex. 1004 at 8:51-67.

5. [3E] "and translating the keywords and any parameters into associated attributes of the CLI command."

Gorthy in combination with Froyd discloses this limitation. Clark Decl. ¶ 91; s*ee also* discussion of claim element 1C. As discussed above, Gorthy discloses a system that translates XML-based configuration commands having CLI syntax to CLI-based commands using an XML schema. *See* Ex. 1003 at 3:9-16. The XML configuration schema is used to "reformat the XML-based command into a proper CLI format." *Id.* at 3:15-16. A POSA would understand that this conversion process involves identifying keywords and any parameters contained in XML tags, and then translating those keywords and parameters into associated attributes of the

CLI commands. Clark Decl. ¶ 91; s*ee also* discussion of claim element 1C; Ex. 1004 at 8:51-67.

D. Claim 4

1. "The method of claim 1, wherein the output message in the XML format having the CLI syntax includes data formatted in accordance with an XML data model of CLI rules and behaviors enforced by an internetwork operating system (IOS) command line interface (CLI) parser subsystem."

Gorthy in combination with Courtney, Froyd, and the JUNOScript Guide discloses this claim. Clark Decl. ¶¶ 92-93. As discussed above with respect to claim element 1D, Courtney discloses a system that translates an output message in a CLI format into an XML format. *See* Ex. 1002 at 2:40-49; *see also* Ex. 1003 at 4:48-51. For instance, Courtney discloses a schema that "can be directly used to generate an XML document that represents the configuration of the particular network device." Ex. 1002 at 3:12-14. This "schema can include a standard representation of the command structure for a particular type of network device" (*id.* at 3:1-3), and Courtney specifically discloses that Cisco routers may be used. Courtney teaches, for instance, that "one schema could contain a representation of the command structure for all model 7500 Cisco™ routers using OS version 12.1, and another schema could contain a representation of the command structure routers using OS version 12.2." *Id.* at 3:3-7; *see also* Clark Decl. ¶ 92. This schema ensures

that translated commands in XML format will be formatted in accordance with proper CLI rules and behaviors. Clark Decl. ¶ 92.

Moreover, as noted above with respect to claim 2, the term "internetwork operating system (IOS)" is a term coined by Cisco that refers to operating system software that runs on Cisco routers. Clark Decl. ¶ 93. Therefore, a POSA would understand that the CLI rules and behaviors, when designed to be used with Cisco routers, are enforced by a component of the internetwork operating system (IOS) (*i.e.*, a command line interface (CLI) parser subsystem). *Id.*

E. Claim 5

1. [5A] "The method of claim 1, wherein the translation of the output message from the CLI format into the XML format having the CLI syntax comprises:"

Gorthy in combination with Courtney, Froyd, and the JUNOScript Guide discloses the method of claim 5, as discussed below. Clark Decl. at ¶ 94.

2. [5B] "parsing the output message to identify at least one CLI token;"

This element is identical to the corresponding requirement in claim element 1D. Thus, Gorthy in combination with Courtney, Froyd, and the JUNOScript Guide discloses parsing the output message to identify at least one CLI token for the reasons set forth above. Clark Decl. ¶ 95; s*ee also* discussion of claim 1D. For instance, Courtney discloses that one of the first steps in a conversion is to "iden-tif[y] each initial command within each configuration line" from the configuration

of the device. Ex. 1002 at 7:17-21. Since such initial commands would be CLI tokens, Courtney's system parses the file in order to identify at least one such CLI token. Clark Decl. ¶ 95.

3. [5C] "accessing a stored mapping of CLI tokens-to-XML values;"

As discussed above with respect to claim element 1D, Courtney discloses the use of a "schema [that] can be directly used to generate an XML document that represents the configuration of the particular network device." Ex. 1002 at 3:12-14. A POSA would understand that when converting between CLI tokens and XML values, the mapping between the two must be stored, even if only in a transitory manner, to accomplish the desired conversion. Clark Decl. ¶ 96. A POSA would also appreciate that, in order to perform the desired conversion, the system would access the stored mapping of CLI tokens-to-XML values. *Id.*; *see also* discussion of claim element 1D. A POSA would recognize that the JUNOScript API, for example, must access a mapping of CLI tokens-to-XML values when the API is instructed to output the results of a CLI command in XML, such as by using the "display xml" command, rather than the default ASCII-formatted text. Clark Decl. ¶ 96; *see also* Ex. 1005 at 17, 37.

4. [5D] "translating each CLI token of the output message into a corresponding XML value, in accordance with said stored mapping;"

This element is substantively identical to the corresponding requirement in claim element 1D of "translating each CLI token of the output message into a corresponding XML value according to a stored mapping of CLI tokens-to-XML values." As discussed above, Courtney discloses the use of a "schema [that] can be directly used to generate an XML document that represents the configuration of the particular network device." Ex. 1002 at 3:12-14. A POSA would understand that when converting between CLI tokens and XML values, the mapping between the two must be stored, even if only in a transitory manner, to accomplish the desired conversion. Clark Decl. ¶ 97; *see also* discussion of claim element 1D. A POSA would recognize that the JUNOScript API, for example, translates the output message of a CLI command according to stored mapping of CLI tokens-to-XML values when using the "display xml" command. Clark Decl. ¶ 97; *see also* discussion of claim element 1D; *see also* Ex. 1005 at 17, 37.

5. [5E] "and generating the output message in the XML format having the CLI syntax with the XML values."

This element is substantially identical to the corresponding requirement in claim element 1D. Thus, Gorthy in combination with Courtney, Froyd, and the JUNOScript Guide discloses this limitation for the reasons set forth above. Clark Decl. ¶ 98; *see also* discussion of claim element 1D. For instance, Courtney dis-

closes a converter that translates an output message in a CLI format to an XML format (*see* Ex. 1002 at 2:40-45), and, in view of the JUNOScript Guide, one of skill in the art would appreciate that one sort of output message that can be translated is an output message providing the current configuration of the system. *See* Ex. 1003 at 4:48-51; *see also* Ex. 1002 at 4:48-49; Ex. 1005 at 37. Messages output in response to such commands requesting configuration information regarding a system can thereafter be converted into XML format using a schema. Ex. 1002 at 3:12-14 ("In certain embodiments, this schema can be directly used to generate an XML document that represents the configuration of the particular network device."). As discussed above, when translating from CLI to XML, the most obvious and sensible approach would be to retain the same syntax as the output message being translated, thereby creating a message in an XML format with the CLI syntax. Clark Decl. ¶ 98. Indeed, the output messages illustrated in the JUNOScript Guide reflect this approach (*see, e.g.*, Ex. 1005 at 17, 37) and would only further confirm to one of ordinary skill in the art that this was the most obvious solution. Clark Decl. ¶ 98; *see also* discussion of claim element 1D.

F. Claim 6

1. [6A] "A computer-usable memory device having computer-readable program code embedded therein for causing a computer system to:"

Claim 6 is substantially identical to claim 1, except that it is drafted as a computer-usable memory device for causing a system to perform a recited method. Because the claimed method to be performed by this computer-usable memory device is substantially the same as claim 1, this claim is invalid for the reasons articulated above. Clark Decl. ¶ 99. Accordingly, Petitioners incorporate by reference and respectfully refer the Board to the discussion of claim 1 above.

Gorthy in combination with Courtney, Froyd, and the JUNOScript Guide discloses a computer-usable memory device having computer readable program code embedded therein for causing a system to perform the steps set forth in the elements below. *Id.* ¶ 100. For example, the invention of Gorthy is claimed as a computer-readable storage medium (*i.e.*, memory device) with program instructions (*i.e.*, computer-readable program code) stored on it for performing the claimed method. Ex. 1003 at 12:4-26. Moreover, Froyd notes that its invention "may be implemented by a processing unit in a digital processing system, which executes sequences of computer program instructions which are stored in a memory which may be considered to be a machine-readable storage media." Ex. 1004 at 16:49-54. Moreover, Froyd explains that the memory device "may be ran-

dom access memory, read only memory, a persistent storage memory, such as mass storage device or any combination of these devices." *Id.* at 16:54-57. And Froyd confirms that when the code is executed, it "causes the processing unit to perform operations according to" the invention. *Id.* at 16:57-59; *see also id.* at 18:9-67 (claims 11-20). Likewise, Courtney discloses that its invention can be carried out using "a plurality of instruction stored on [a] storage device," where "the plurality of instructions [are] configured to cause the processor" to carry out the steps of the invention. Ex. 1002 at 10:3-5 (claim 12).

2. [6B] "receive an input command requesting an operation be performed by a routing system, wherein the input command is configured in an extensible markup language (XML) format having a command line interface (CLI) syntax with CLI keywords sequenced according to configuration rules for CLI commands;"

This limitation is substantially identical to that of claim 1B, and the above analysis of that claim accordingly applies to this limitation. Clark Decl. ¶ 101; *see also* discussion of claims 1B.1 and 1B.2. Claim 6B lacks the requirement of claim 1B.1 that the input command be received "with a command line interface (CLI) parser," and the analysis of the narrower claim 1B.1 limitation thus applies here. Clark Decl. ¶ 101. Claim 6B requires that the input command "request[] an operation be performed by a routing system," whereas claim 1B.1 requires the input command be "configured to request an operation be performed by a routing system." *Id.* This distinction is not relevant to the analysis of claim 1B.1 and 1B.2,

as the XML configuration command of Gorthy is both "configured to request" that an operation be performed by a routing system and, moreover, that the command actually "requests" the operation be performed. *Id.*; *see also* Ex. 1003 at 3:9-19, 6:3-15. For these and the reasons discussed above as to Claims 1B.1 and 1B.2, Gorthy discloses this limitation.

3. [6C] "translate the input command from the XML format having the CLI syntax into a CLI command, wherein the routing system is configured to execute the CLI command and perform the operation, and wherein the computer system is further configured to translate the input command by identifying at least one XML tag that includes an XML parameter to indicate the XML tag includes one or more CLI keywords, extracting the one or more CLI keywords from the input command, and arranging the one or more CLI keywords into the CLI command according to the CLI syntax of the input command;"

This limitation is substantially identical to that of claim 1C, and the above analysis of that claim accordingly applies to this limitation. Clark Decl. ¶ 102; *see also* discussion of claims 1C.1, 1C.2, 1C.3. Claim 6C lacks the requirement of claim 1C.1 that the input command be translated "with the CLI parser," and the analysis of the narrower claim 1C.1 limitation thus applies here. Clark Decl. ¶ 102. Claim 6C requires that the routing system be "configured to execute the CLI command and perform the operation," whereas claim 1C.2 requires that the CLI command "when executed, is configured to prompt the routing system to perform the operation" and that the "routing system is configured to perform the operation re-

sponsive to the execution of the CLI command." *Id.* The analysis of claim 1C.2 applies to claim 6C because, in both claims, Gorthy's input command, once converted to a CLI command, is executed by the routing system, which causes the routing system to perform the operation. *Id.*; s*ee also* Ex. 1003 at 3:9-19, 6:3-15, 6:21-26; *see id.* at Fig. 8. Whether claimed as the input command that prompts an operation to be performed when executed or the routing system that executes the command, prompting the operation to be performed, Gorthy discloses each limitation. Clark Decl. ¶ 102. For these and the reasons discussed above as to claims 1C, Gorthy discloses this limitation.

4. [6D] "translate an output message from a CLI format into XML format having the CLI syntax, wherein the output message is generated in the CLI format by the routing system responsive to the performance of the operation, and wherein the translating includes parsing the output message to identify at least one CLI token, translating each CLI token of the output message into a corresponding XML value according to a stored mapping of CLI tokens-to-XML values, and generating the output message in the XML format with the XML values;"

This limitation is substantially identical to that of claim 1D, and the above analysis of that claim accordingly applies to this limitation. Clark Decl. ¶ 103; *see also* discussion of claim 1D. The difference between these limitations is that claim 1D specifies that the output message is "generated in response to performance of the operation," whereas claim 6D indicates the output message is "generated in the CLI format by the routing system responsive to the performance of the opera-

tion." Clark Decl. ¶ 103. The analysis of claim 1D applies because the output message of Courtney is generated by the routing system (*e.g.*, a Cisco router) in CLI format in response to the operation. *Id.*; *see also* Ex. 1002 at 4:48-49; *see also* Ex. 4:48-51. Likewise, the output message generated by the JUNOScript API, running on a Juniper Networks router, is natively in a CLI format in response to performing the operation. Clark Decl. ¶ 103; Ex. 1005 at 37. For these and the reasons discussed above as to claim 1D, Gorthy discloses this limitation.

5. [6E] "and transmit the output message in the XML format having the CLI syntax to a remote device external from the routing system."

This limitation is identical to that of claim 1E, except that claim 1E uses the gerund, "transmitting," rather than claim 6E's infinitive "[to] transmit." Clark Decl. ¶ 104. Accordingly, the analysis of claim 1E applies to this limitation. *See* discussion of claim 1E. For the reasons identified above as to claim 1E, Gorthy in combination with Froyd discloses this limitation.

G. Claim 7

1. "The computer-usable memory device of claim 6, wherein the input command is formatted in accordance with an XML schema of CLI rules and behaviors enforced by an internetwork operating system (IOS) command line interface (CLI) parser subsystem."

Claim 7 recites the same method as claim 2, but is drafted in the form of a computer-usable memory device with embedded code for performing the claimed steps. Clark Decl. ¶ 105. As noted above with respect to claim element 6A, both

Courtney and Froyd expressly disclose such a computer-usable memory device. *See* Ex. 1004 at 16:49-59; *see also id.* at 18:9-67 (claims 11-20); Ex. 1002 at 10:3-5 (claim 12). Moreover, a POSA would recognize that an obvious and conventional way to allow a method to be practiced is to place code that would carry out the method onto a computer-usable memory device. Clark Decl. ¶ 105.

Because the method performed using the computer-readable memory device of claim 7 is identical to that recited in claim 2, this claim is disclosed by Gorthy in combination with Courtney, Froyd, and the JUNOScript Guide for the reasons discussed above. *Id.* at ¶ 106; *see* discussion accompanying claim 2.

H. Claim 8

Claim 8 recites the same method as claim 3, but is drafted in the form of a computer-usable memory device with embedded code for performing the claimed steps. As noted above with respect to claim element 6A, both Courtney and Froyd expressly disclose such a computer-usable memory device. *See* Ex. 1004 at 16:49-59; *see also id.* at 18:9-67 (claims 11-20); Ex. 1002 at 10:3-5 (claim 12). Moreover, a POSA would recognize that an obvious and conventional way to allow a method to be practiced is to place code that would carry out the method onto a computer-usable memory device. Clark Decl. ¶ 107.

Because the method performed using the computer-readable memory device of claim 8 is identical to that recited in claim 3, this claim is disclosed by Gorthy in

combination with Courtney, Froyd, and the JUNOScript Guide for the same reasons discussed above. *Id.*; *see also* discussion of claim 3. In lieu of repeating the same arguments for claim 8B-8E, Petitioner incorporates by reference and respectfully refers the panel to the discussion of claim 3B-3E above.

I. Claim 9

1. "The computer-usable memory device of claim 6, wherein the output message in the XML format having the CLI syntax includes data formatted in accordance with an XML data model of CLI rules and behaviors enforced by an internetwork operating system (IOS) command line interface (CLI) parser subsystem."

Claim 9 recites the same method as claim 4, but is drafted in the form of a computer-usable memory device with embedded code for performing the claimed steps. As noted above with respect to claim element 6A, both Courtney and Froyd expressly disclose such a computer-usable memory device. *See* Ex. 1004 at 16:49-59; *see also id.* at 18:9-67 (claims 11-20); Ex. 1002 at 10:3-5 (claim 12). Moreover, one of ordinary skill in the art would recognize that an obvious and conventional way to allow a method to be practiced is to place code that would carry out the method onto a computer-usable memory device. Clark Decl. ¶ 113.

Because the method performed using the computer-readable memory device of claim 9 is identical to that recited in claim 4, this claim is disclosed by Gorthy in combination with Courtney, Froyd, and the JUNOScript Guide as discussed above. *Id.* at ¶ 114. For instance, as explained with respect to claim 4, Courtney discloses

a schema that "can be directly used to generate an XML document that represents the configuration of the particular network device." Ex. 1002 at 3:12-14. This "schema can include a standard representation of the command structure for a particular type of network device" (*id.* at 3:1-3), and Courtney specifically discloses that Cisco routers may be used. *Id.* at 3:3-7. Because Courtney's system is described with regard to Cisco routers, one of ordinary skill would understand that the CLI rules and behaviors, when designed to be used with Cisco routers, are enforced by a component of the internetwork operating system (IOS) (*i.e.*, a command line interface (CLI) parser subsystem). Clark Decl. ¶ 114.

J. Claim 10

Claim 10 recites the same method as claim 5, but is drafted in the form of a computer-usable memory device with embedded code for performing the claimed steps. As noted above with respect to claim element 6A, both Gorthy and Froyd disclose such a computer-usable memory device. *See* Ex. 1004 at 16:49-59; *see also id.* at 18:9-67 (claims 11-20); Ex. 1002 at 10:3-5 (claim 12). Moreover, one of ordinary skill in the art would recognize that an obvious and conventional way to allow a method to be practiced is to place code that would carry out the method onto a computer-usable memory device. Clark Decl. ¶ 115. Because the method performed using the computer-readable memory device of claim 10 is identical to that recited in claim 5, this claim is disclosed by Gorthy in combination with Courtney,

Froyd, and the JUNOScript Guide for the same reasons discussed above. *Id.*; *see also* discussion of claim 5. In lieu of repeating the same arguments for claim 10B-10E, Petitioner incorporates by reference and respectfully refers the panel to the discussion of claim 5B-5E above.

IX. CONCLUSION

Petitioner respectfully requests institution of an *inter partes* review and cancellation of claims 1-10 of the '886 patent.

Respectfully submitted,

DATED: November 24, 2015

/s/ Leo Lam

Leo Lam, Registration No. 38,528
Counsel for Petitioner, Arista Networks, Inc.

Keker & Van Nest LLP
633 Battery Street
San Francisco, CA 94111
(415) 391-5400 (telephone);
(415) 397-7188 (fax)

CERTIFICATION OF SERVICE (37 C.F.R. §§ 42.6(e), 42.105(a))

The undersigned hereby certifies that a copy of the foregoing **PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,953,886**, the accompanying Power of Attorney, list of exhibits for Petition for *Inter Partes* Review, and all associated exhibits were served in their entireties on the following parties via FedEx®:

Cisco c/o Leon R. Turkevich
Manelli Selter PLLC
10560 Main Street, Suite PH6
Fairfax, VA 22030
Address of record as listed on PAIR

Sean Sang-Chul Pak
John M. Neukom
Matthew Cannon
Quinn Emanuel Urquhart & Sullivan LLP
50 California Street, Floor 22
San Francisco, CA 94111
seanpak@quinnemanuel.com
johnneukom@quinnemanuel.com
Cisco-Arista@quinnemanuel.com
Additional address known to Petitioner as likely to effect service

Adam R. Alper
Kirkland & Ellis LLP
555 California Street
San Francisco, CA 94104
aalper@kirkland.com

Additional address known to Petitioner as likely to effect service

/s/ *Leo Lam*
Leo Lam, Registration No. 38,528
Counsel for Petitioner, Arista Networks, Inc.

Keker & Van Nest LLP
633 Battery Street
San Francisco, CA 94111
(415) 391-5400 (tel); (415) 397-7188 (fax)